

### **Remarks**

In the above captioned Office Action claims 1-55 have been rejected. Claims 1, 16, 23, 29, 44, and 51 have been amended.

### **New Claim**

Claim 56 has been added to the instant application. Applicants submit that no new matter has been added. In particular, claim 56 is exemplary and claims:

The method of claim 1, wherein each of the current and next states is a selected one of:

the lock is not held and there no threads waiting to access the shared resource,  
the lock is held and there are no threads waiting to access the shared resource,  
the lock is held and there is one thread waiting to access the shared resource,  
and  
the lock is held and there are at least two threads waiting to access the shared resource.

These states are disclosed by the specification on page 6 lines 3-6, page 12 lines 5-6 and figure 2 of the specification.

### **Rejections under 35 U.S.C. § 101**

In "Claim Rejections – 35 USC §101" on page 2 of the above-identified Office Action, the Examiner rejects Claims 16-28 as being directed to non-statutory subject matter. More specifically, the Examiner asserts that the apparatus of claims 16-22 claim a lock and a lock acquirer; and claims 23-28 claim a lock and a lock releaser. Further, the Examiner asserts the specification sets forth no hardware for these features and therefore, Claims 16-28 appear to be software per se.

In response, Applicants have amended claims 16, 23, 29, 44 and 51, obviating the Examiner's rejections, support for the amendments can be found on page 17 of the specification, and no new matter has been added.

**Claim Rejections under 35 U.S.C. § 102(e)**

On page 3 item 4 of the above-identified Office Action the Examiner states claims 1-55 are rejected under 35 U.S.C. § 102(e) as being fully anticipated by U.S. Patent Application Publication No. 2003/0200457 A1 to Auslander et al (hereinafter "Auslander").

Claim 1 is exemplary and claims a method of managing a lock utilized by a plurality of threads executing on a computing device to coordinate access to a shared resource, the method comprising:

selecting by one of the threads, an action to be performed by the thread upon the lock, wherein the action is selected from a group comprising:

acquiring the lock,  
trying to acquire the lock, and  
releasing the lock;

asynchronously querying and receiving a current state of the lock by the thread,  
the lock being considered to be in any one of at least four states in any point in time,  
and each state is represented by a multi-part state value conveying multiple information;

speculatively determining by the thread, the next state of the lock, where the next

state is the state of the lock if the thread proceeds to perform the selected action and the thread is successful; and

attempting to perform by the thread, the selected action to transition the lock from the current state to the speculatively determined next state.

Claim 1, as amended claims a method of managing a lock utilized by a thread, the lock being able to be in any one of at least four states at any point in time, each state being represented by a multi-state value conveying multiple information. (figure 4 of the instant application)

Auslander teaches an "enhancement to the MCS lock for increased functionality and improved programmability" (Auslander, title). This is achieved by using a two pointer mechanism, this eliminates the need for pre-allocation of the qnode structure, and the memory for this qnode structure is allocated from the stack by the compiler. Page 3, paragraph 22 of Auslander, describes a routine, used to manipulate pointers to achieve the enhancement to the MCS lock, this includes declaring a structure with two pointers that are initialized as null representing a qnode, the first qnode being the lock. As further threads try to acquire the lock they are enqueued with similar qnode structures and wait their turn for the lock. "The code then spins until it has been notified it has the lock (line 035 of FIG.4). Upon releasing a lock held by multiple threads, where the tail pointer is not pointing to the head, a first thread waits for it's head pointer (this is implicitly the head pointer of the lock at this point) to become non-null (line 110 of the acquire routine in FIG. 4) (remember the crucial order described in the acquire)

and then grants the lock to a second thread pointed to by its head.” (Auslander, page 3 Para 22) Auslander teaches a method where a thread waits by spinning until the lock is available. Once the lock becomes available the qnode in possession of the lock notifies the next qnode in line allowing the next thread to gain access to the lock. Auslander outlines a method where the thread waits until it is notified that the lock is free and then the thread is moved to the head of the queue and the lock is reset. “The granting of a lock occurs by setting the waiting bit of next thread in line ... When one thread is granted a lock via another thread releasing the lock.” (Auslander, page 3 para 22). Auslander teaches a method where the thread can be in any one of three states at any point in time. (Auslander, Figure 2). The three states being either “unlocked”, “locked no waiters” and “locked with waiters” these three states being established by the permutations of where the pointers point from either the head or tail of the lock. The head and tail pointers of the lock can either point to “NULL”, “HEAD”, or “TAIL no other information is held by the thread, Auslander does not describe a lock which contains additional information to that, that establishes the condition of the thread. Auslander does not describe a lock that can be “considered to be in any one of four states represented by a multi part state value conveying multiple information;” (instant application Claim 1)

Therefore, Auslander does not claim “...asynchronously querying and receiving a current state of the lock by the thread, the lock being considered to be in any one of four states in any point in time, and each state is represented by a multi-part state value conveying multiple information;” (Claim 1, instant application). Auslander therefore

does not claim each and every element of claim 1 and therefore, claim 1 is not anticipated. Reconsideration and allowance is respectfully requested.

Independent claims 16, 23, 29, 44 and 51 contain similar recitations to claim 1 and, for at least the reason cited above, are also not anticipated. Reconsideration and allowance is respectfully requested.

Claims 2-15 depend from claim 1, incorporating its recitations; Claims 17-22 depend from claim 16, incorporating its recitations; Claims 24-28 depend from claim 23, incorporating its recitations; Claims 30-43 depend from claim 29, incorporating its recitations; Claims 45-50 depend from claim 44, incorporating its recitations; and Claims 52-55 depend from claim 51, incorporating its recitations and are for at least the reasons cited above, not anticipated and therefore patentable over Auslander. Reconsideration and allowance is respectfully requested.

In view of the foregoing, the Examiner is respectfully requested to withdraw the rejections under 35 U.S.C. § 102(e) in item 3 "Claim Rejections – 35 USC § 102" on page 2 of the above-identified office action.

#### **Claim 56**

Notwithstanding the above, Applicants submit the following additional reason for the allowability of claim 56. Claim 56 recites the method of claim 1, wherein each of the current and next states is a selected of:

the lock is not held and there no threads waiting to access the shared resource,  
the lock is held and there are no threads waiting to access the shared resource,

the lock is held and there is one thread waiting to access the shared resource,  
and  
the lock is held and there are at least two threads waiting to access the shared  
resource.

Auslander can resolve the current state of the lock by determining the values  
pointed to by the two pointers of the lock qnode structure. As discussed above,  
Auslander discloses an "enhancement to the MCS lock for increased functionality and  
improved programmability", Auslander discloses a method where the lock can be in one  
of three states. In contrast Claim 56 of the instant application recites a lock with at least  
four particularized states. Auslander does not teach a method with a multi-part state  
value conveying multiple information, including a flag value that contains information  
about the current state of the lock, which is a selected one of four enumerated states.

Applicants therefore submit that Auslander fails to teach or suggest each and  
every element of claim 56 and in fact teaches away from this limitation. Applicants  
submit that claim 56, for at least this additional reason, is nonobvious and therefore  
patentable over Auslander.

**Conclusion**

In conclusion, claims 1-56 are in condition for allowance. Early issuance of Notice of Allowance is respectfully requested.

The Commissioner is hereby authorized to charge shortages or credit overpayments to Deposit Account No. 500393.

Respectfully submitted,  
SCHWABE, WILLIAMSON & WYATT, P.C.

Dated: July 31, 2007

/William A. Holmes/  
William A. Holmes  
Registration No. 59,024

Pacwest Center, Suite 1900  
1211 SW Fifth Avenue  
Portland, Oregon 97204  
Telephone: 503-222-9981